

Lyndon Words and Short Superstrings

Marcin Mucha*
mucham@mimuw.edu.pl

March 6, 2013

Abstract

In the shortest superstring problem, we are given a set of strings $\{s_1, \dots, s_k\}$ and want to find a string that contains all s_i as substrings and has minimum length. This is a classical problem in approximation and the best known approximation factor is $2\frac{1}{2}$, given by Sweedyk [19] in 1999. Since then no improvement has been made, however two other approaches yielding a $2\frac{1}{2}$ -approximation algorithms have been proposed by Kaplan et al. [10] and recently by Paluch et al. [16] — both based on a reduction to maximum asymmetric TSP path (MAX-ATSP-PATH) and structural results of Breslauer et al. [5].

In this paper we give an algorithm that achieves an approximation ratio of $2\frac{11}{23}$, breaking through the long-standing bound of $2\frac{1}{2}$. We use the standard reduction of SHORTEST-SUPERSTRING to MAX-ATSP-PATH. The new, somewhat surprising, algorithmic idea is to take the better of the two solutions obtained by using: (a) the currently best $\frac{2}{3}$ -approximation algorithm for MAX-ATSP-PATH and (b) a naïve cycle-cover based $\frac{1}{2}$ -approximation algorithm. To prove that this indeed results in an improvement, we further develop a theory of string overlaps, extending the results of Breslauer et al. [5]. This theory is based on the novel use of Lyndon words, as a substitute for generic unbordered rotations and critical factorizations, as used by Breslauer et al.

1 Introduction

The Shortest Superstring Problem In the SHORTEST-SUPERSTRING problem we are given a set of strings $\{s_1, \dots, s_k\}$ and want to find a string that contains all s_i as substrings and has minimum length. The problem has several applications including data compression [8, 18] and DNA sequencing [13, 14, 17, 24]. In the latter, one attempts to reconstruct a DNA molecule, which is a string over the alphabet $\{A, C, G, T\}$, based on a massive set of short fragments. These fragments (i.e. substrings) of the molecule can be obtained by *sequencing*. The reconstruction problem can be viewed as a shortest superstring problem based on the premise that the original molecule is a superstring of all the fragments, and that shorter superstrings should in general be more similar to the original.

Previous Results Since SHORTEST-SUPERSTRING is NP-hard [8, 9] and even MAX-SNP-hard [4, 23], the best we can hope for in terms of approximation is a constant factor. A lot of effort went into designing approximation algorithms for the problem, Table 1 summarizes these developments. Note that the last two results, by Kaplan et al. [10] and Paluch et al. [16]

*This work was partially supported by the ERC StG project PAAI no. 259515

Authors	Date	Factor
Li [14]	1990	$O(\log(n))$
Blum, Jiang, Li, Tromp, Yannakakis [4]	1991	3
Teng, Yao [21]	1993	$2\frac{8}{9}$
Czumaj, Gasieniec, Piotrów, Rytter [7]	1994	$2\frac{5}{6}$
Kosaraju, Park, Stein [12]	1994	$2\frac{50}{63}$
Armen, Stein [1]	1995	$2\frac{3}{4}$
Armen, Stein [2]	1996	$2\frac{2}{3}$
Breslauer, Jiang, Jiang [5]	1997	$2\frac{25}{42}$
Sweedyk [19]	1999	$2\frac{1}{2}$
Kaplan, Lewenstein, Shafrir, Sviridenko [10]	2005	$2\frac{1}{2}$
Paluch, Elbassioni, van Zuylen [16]	2012	$2\frac{1}{2}$

Table 1: Previous results for the SHORTEST-SUPERSTRING problem.

do not improve the approximation factor. They both give $\frac{2}{3}$ -approximation algorithms for the related MAX-ATSP-PATH problem. Using a black-box reduction due to Breslauer et al. [5], these give $2\frac{1}{2}$ -approximation algorithms for SHORTEST-SUPERSTRING. Both, especially the one due to Paluch et al., are significantly simpler than the original result of Sweedyk.

Parallel to these developments, some progress has been made towards resolving the *Greedy Superstring Conjecture* (see [18, 20, 22]), which says that the greedy approach of repeatedly picking the two strings that overlap the most and gluing them together until only a single string remains, is actually a 2-approximation. Blum et al. [4] showed that the greedy algorithm gives a 4-approximation, and Kaplan et al. [11] improved this to $3\frac{1}{2}$.

Our Results/Techniques In this paper we develop several results that describe the structure of the overlaps of a collection of strings. Our results can be viewed as an extension of the framework introduced by Breslauer et al. [5]. However, while Breslauer et al. use generic unbordered rotations and critical factorizations, we construct ours by using Lyndon words. It turns out that the added control we gain in this way allows for much more precise structural analysis of string overlaps.

We use these results to obtain a $2\frac{11}{23}$ -approximation for SHORTEST-SUPERSTRING, and therefore break a long-standing bound of $2\frac{1}{2}$.

The basic idea of our approach is the following. For two strings u, v , let the *overlap* of u and v , denoted $ov(u, v)$, be the longest suffix of u which is also a prefix of v . The *overlap graph* of a set of strings S is a complete directed graph on S with edge weights equal to lengths of corresponding overlaps.

Blum et al. [4] show how approximating SHORTEST-SUPERSTRING for a set of strings S can be reduced to approximating the problem of finding a longest path in the overlap graph of a certain auxiliary set of strings $R(S)$, called *representative strings*. The performance of the resulting algorithm depends on how well we can bound the *overlap loss* in the longest path approximation.

This bound can essentially be improved in two ways: by using a better approximation algorithm for the longest path problem in directed graphs (MAX-ATSP-PATH), or by providing a better bound on the overlap of the optimum path. For the first direction, Kaplan et al. [10] and Paluch et al. [16] both give $\frac{2}{3}$ -approximation for MAX-ATSP-PATH, which is the

best known. For the second, the bounds given by Breslauer et al. [5] are essentially tight.

In this paper we propose a third way to improve by joining the two objectives. Note that one can approximate MAX-ATSP-PATH by finding a maximum weight cycle cover, removing the lightest edge on each cycle, and then joining the resulting paths with arbitrary edges. This naïve algorithm only gives $\frac{1}{2}$ -approximation, significantly weaker than $\frac{2}{3}$, the tight case being *balanced* 2-cycles. We observe however, that with a careful choice of representative strings $R(S)$, if the bounds given by Breslauer et al. are nearly tight, the cycles in the maximum weight cycle cover are far from balanced. So far in fact, that choosing the better of the two solutions: one given by a $\frac{2}{3}$ -approximation algorithm, and one given by our naïve algorithm, results in an approximation algorithm for SHORTEST-SUPERSTRING with ratio strictly smaller than $2\frac{1}{2}$.

It is worth noting that, similarly to the approach of Breslauer et al., our algorithm is a black-box reduction from SHORTEST-SUPERSTRING to MAX-ATSP-PATH. Therefore, any improvements on the approximation factor for the latter will yield an improvement for the former.

Organization of the Paper The paper is organized as follows. In Section 2 we recall some facts regarding the properties of strings and their overlaps, as well as the standard approach to shortest string approximation. In Section 3 we describe the new algorithm and analyse its approximation factor. This analysis relies on Theorem 3.2, which is the main technical result of this paper. The remaining part of the paper is devoted to proving this theorem.

In Section 4 we present some general bounds concerning overlaps of strings. We believe they might be of independent interest. In Section 5 we use these bounds to prove the main theorem. Since the proof is a rather long and detailed case analysis, to facilitate understanding of the basic ideas of the paper, in Subsection 5.2 we give a simple proof of a weaker version of the main theorem. This version still gives an approximation factor smaller than $2\frac{1}{2}$.

Finally in Section 6 we show that Theorem 3.2 is essentially tight. We also briefly discuss reasons why using our bounds to improve the analysis of the greedy algorithm might be difficult.

2 Preliminaries

In this section we recall some definitions, results and ideas concerning basic properties of strings. For a more extensive exposition the reader should consult any of the standard textbooks on combinatorics on words, e.g. the excellent monograph by Lothaire [15].

We also describe the standard framework for SHORTEST-SUPERSTRING approximation. Our presentation mostly follows that of Breslauer et al. [5]. Note however, that instead of generic critical factorizations we use nice rotations, introduced at the end of Subsection 2.1. This requires almost no changes in the framework, except for the proof of Lemma 2.7, which we provide.

2.1 Stringology

Basic concepts For a string v , we will use $v[i]$ to denote the i -th letter of v , and $v[i, j]$ to denote the substring of v consisting of letters i, \dots, j . We will use vu to denote concatenation of v and u , and v^k to denote the concatenation of k copies of v . We will also use v^∞ to denote

the semi-infinite string $vvv\dots$. Any representation of $w = uv$ as a concatenation of two (not necessarily nonempty) strings is called a *factorization* of w . The factorization is *nontrivial* if both u and v are nonempty.

For a string w of length n , any integer $1 \leq p \leq n$ is a *period* of w if $w[i] = w[i + p]$ for all $1 \leq i \leq n - p$. Note that w always has at least one period, that is its length. The smallest period of w is called *the minimum period* of w or simply *the period* of w , and denoted $p(w)$.

A string w is *primitive* if there is no v such that $w = v^k$ with $k \geq 2$.

A string z is called a *rotation* of w if there exists a factorization $w = uv$ such that $z = vu$. In that case we also say that z is a rotation *starting at position* $|u| + 1$, or that $|u| + 1$ is z 's starting position in w . It is easy to see that if z is a rotation of w , then z is primitive iff w is. It is also a standard fact that if w is primitive, then any rotation corresponding to a nontrivial factorization of w is different from w . More generally, for a primitive w and two different factorizations $w = u_1v_1$ and $w = u_2v_2$, the rotations v_1u_1 and v_2u_2 are different. It follows that for primitive w , every rotation of w has a unique starting position in w .

We say that two strings are *equivalent* if one is a rotation of the other. Otherwise they are *non-equivalent*.

We will assume a fixed order on the alphabet. This order induces a standard lexicographical order on the set of strings. We use $u \prec v$ to denote that u is lexicographically smaller than v , and $u \preceq v$ to denote that u is smaller or equal to v .

Let w be a primitive string and consider the order induced by \prec on all rotations of w . Let w_{\min} and w_{\max} be the minimal and maximal rotations in this order. Also, denote by $i_{\min}(w)$ and $i_{\max}(w)$ the starting positions of w_{\min} and w_{\max} in w . Moreover, let $p_{\min}(w)$ and $p_{\max}(w)$ be strings such that $w_{\min} = p_{\min}(w)p_{\max}(w)$ and $w_{\max} = p_{\max}(w)p_{\min}(w)$.

Lemma 2.1. *Let w , $|w| \geq 2$ be a primitive string. Then both $p_{\min}(w)$ and $p_{\max}(w)$ are nonempty. In other words, $w_{\min} \neq w_{\max}$.*

Proof. Since $|w| \geq 2$ and w is primitive, it contains at least 2 different letters. It follows that w_{\min} and w_{\max} start with different letters, and so they are different strings. In particular both $p_{\min}(w)$ and $p_{\max}(w)$ are nonempty. \square

The following property of w_{\min} and w_{\max} is implicitly used by Crochemore et al. [6].

Lemma 2.2. *Let w , $|w| \geq 2$ be a primitive string. Then w_{\max} is the only rotation of w that starts with $p_{\max}(w)$, and w_{\min} is the only rotation that starts with $p_{\min}(w)$.*

Proof. We will prove the claim for $p_{\max}(w)$, the other part of the proof is analogous. Suppose that there is a rotation $z = p_{\max}(w)v$ of w , other than $w_{\max} = p_{\max}(w)p_{\min}(w)$, that starts with $p_{\max}(w)$. We claim that $z \succ w_{\max}$ which is a contradiction. To see that, notice that they both start with $p_{\max}(w)$, and $v \succeq p_{\min}(w)$ since $p_{\min}(w)$ is a prefix of the minimal rotation of w . The claim follows, since $v \neq p_{\min}(w)$. \square

Borders A nonempty string b is called a *border* of a string w if $w = bu = vb$ for some nonempty u, v . A string is *unbordered* if it has no border. So, a string w is unbordered if w has no proper prefix that is also a suffix of w . The following is a standard fact (see e.g. Proposition 5.1.2 in Lothaire [15]).

Lemma 2.3. *Every primitive string w has a rotation that is unbordered. In particular, w_{\min} and w_{\max} are unbordered.*

Proof. Suppose that w_{\max} has a border, i.e. there exists a proper prefix v of w_{\max} which is also its suffix. Let $|w| = |w_{\max}| = n$ and $|v| = k < n$. Since v is a suffix of w_{\max} , we know that $u = (vw_{\max})[1, n]$ is a rotation of w . We claim that $u \succ w_{\max}$, which is a contradiction. To see this, notice that $u[1, k] = v = w_{\max}[1, k]$ and also $u[k+1, n] = w_{\max}[1, n-k] \succeq w_{\max}[k+1, n]$, since w_{\max} is maximal. So $u \succeq w_{\max}$, but this cannot be an equality since rotations of a primitive string are all different.

The same proof applies to w_{\min} or one can simply notice that it is a maximal rotation in the lexicographical order induced by the reversed order on the alphabet. \square

Remark. A primitive string w such that $w = w_{\max}$ is called a Lyndon word¹ (w.r.t. the particular order on the alphabet that is used to define the lexicographical order). Note that the two rotations appearing in Lemma 2.3 are Lyndon words, and in fact this Lemma is equivalent to saying that Lyndon words are unbordered.

Remark. One of the key ingredients of the results of Breslauer et al. [5] is the notion of a critical factorization and „The Critical Factorization Theorem” (see Césari et al. [25]). Although we do not use them directly, a reader acquainted with Breslauer et al. [5] will realize that they are nevertheless present in our work. In particular $p_{\min}(w)p_{\max}(w)$ and $p_{\max}(w)p_{\min}(w)$ are critical factorizations (this fact was used by Crochemore et al. [6] in their proof of the Critical Factorization Theorem).

Nice rotations Let w , $|w| \geq 2$ be a primitive string. The nice rotation of w is defined to be w_{\max} if $|p_{\max}(w)| \leq |p_{\min}(w)|$, otherwise it is defined to be w_{\min} . Let $\alpha(w) = \min(|p_{\max}(w)|, |p_{\min}(w)|)$. We will call a primitive string *nice* if it is its own nice rotation. Note that if w is nice, then:

- $w = w_{\max}$ and $\alpha(w) = |p_{\max}(w)| \leq |w|/2$, or
- $w = w_{\min}$ and $\alpha(w) = |p_{\min}(w)| < |w|/2$.

In particular we always have $\alpha(w) \leq |w|/2$.

For a nice string w , we call x a w -string if x is a prefix of w^∞ .

2.2 Shortest Superstring Approximation

Basic ideas In the remainder of this paper we assume w.l.o.g. that S contains at least two strings and that no string in S is a substring of another string

For two strings u, v define the *overlap* of u and v , denoted $ov(u, v)$, as the longest suffix of u that is also a prefix of v . Also, define the *prefix* of u w.r.t. v , denoted $pref(u, v)$, as the string x such that $u = xov(u, v)$, i.e. prefix is the part of u that does not overlap v .

The following two directed graphs are good models of how the strings in S overlap with each other. The *overlap graph* of S is a complete directed graph with S as the vertex set, and edge (s_i, s_j) having length $|ov(s_i, s_j)|$. The *prefix graph* (also called the *distance graph*) is defined similarly, only edge (s_i, s_j) now has length $|pref(s_i, s_j)|$.

¹We use term “word” here and not “string” as “Lyndon word” seems to be a well established phrase. In general, the algorithmic community tends to use the term “string” and the combinatorial community uses the term “word”. We decided to follow this rule and use the term “string” with the single exception of “Lyndon word”.

Let $\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle$ be the string $\text{pref}(s_{i_1}, s_{i_2})\text{pref}(s_{i_2}, s_{i_3}) \dots \text{pref}(s_{i_{n-1}}, s_{i_n})s_{i_n}$. Obviously, it is the shortest string containing $s_{i_1}, s_{i_2}, \dots, s_{i_n}$ in that order. Notice that the optimal solution has the form $\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle$ for some ordering $s_{i_1}, s_{i_2}, \dots, s_{i_n}$ of the strings in S .

The length of $\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle$ is equal to

$$|\text{pref}(s_{i_1}, s_{i_2})| + |\text{pref}(s_{i_2}, s_{i_3})| + \dots + |\text{pref}(s_{i_{n-1}}, s_{i_n})| + |\text{pref}(s_{i_n}, s_{i_1})| + |\text{ov}(s_{i_n}, s_{i_1})|,$$

which is the length of the cycle $s_{i_1} \rightarrow s_{i_2} \rightarrow \dots \rightarrow s_{i_n}$ in the prefix graph of S increased by $|\text{ov}(s_{i_n}, s_{i_1})|$. Thus, the length of the shortest TSP tour in the prefix graph of S lowerbounds the length of the shortest superstring.

The above considerations suggest that reduction to asymmetric TSP might be useful in approximating SHORTEST-SUPERSTRING. Unfortunately, the best known approximation algorithm for asymmetric TSP has factor $O\left(\frac{\log n}{\log \log n}\right)$ (see [3]), so this approach is not very useful.

Let us look again at a generic solution $\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle$ and this time express its length in terms of the overlap graph:

$$|\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle| = \sum_{j=1}^n |s_j| - \sum_{j=1}^{n-1} |\text{ov}(s_{i_j}, s_{i_{j+1}})|.$$

The right term in the above expression (the *total overlap* of $\langle s_{i_1}, s_{i_2}, \dots, s_{i_n} \rangle$) is the length of the path s_{i_1}, \dots, s_{i_n} in the overlap graph, so the longest TSP path in the overlap graph corresponds to the optimal solution for SHORTEST-SUPERSTRING. Longest TSP path in a directed graph (called MAX-ATSP-PATH) can be approximated within constant factor. Notice however, that this does not lead to a constant factor approximation for SHORTEST-SUPERSTRING. The problem is that the total overlap of the optimal solution could be very large compared to its length. In that case even a very good approximation algorithm for total overlap might only give mediocre approximation for the length of the superstring.

Two-step reduction to MAX-ATSP-PATH We can avoid the problems described in the previous paragraph by using the following two-step approach introduced by Blum et al. [4]:

1. Find a minimum cycle cover \mathcal{C}_{\min} in the distance graph.
2. For each cycle $C \in \mathcal{C}_{\min}$ construct a representative string $R(C)$ containing all strings in C as substrings, let $R = R(\mathcal{C}_{\min}) = \{R(C) : C \in \mathcal{C}_{\min}\}$.
3. Find a SHORTEST-SUPERSTRING solution for R by reducing to MAX-ATSP-PATH.

The idea here is that the first step groups strings with large overlaps together, so that the overlaps of the strings in R are relatively small, and then the last step actually gives good approximation.

The following series of lemmas and definitions from Blum et al. [4] and Breslauer et al. [5] gives an idea of why this approach works.

For any cycle $C = s_{i_1} \rightarrow s_{i_2} \rightarrow \dots \rightarrow s_{i_k}$ in \mathcal{C}_{\min} let $R(C) = \langle s_{i_1}, s_{i_2}, \dots, s_{i_k}, s_{i_1} \rangle$. Note two interesting features of this definition. First, depending on where we break the cycle we can start $R(C)$ with any of the strings s_{i_1}, \dots, s_{i_k} . Second, $R(C)$ is actually “too long” as it unnecessarily contains two copies of s_{i_1} — this will turn out useful later on.

Let $OPT(S)$ and $OPT(R)$ be the lengths of optimal SHORTEST-SUPERSTRING solutions for S and R .

Lemma 2.4 (Follows from Lemma 2.6 of [5], also implicit in [4]).

$$OPT(R) \leq 2OPT(S).$$

For a cycle $C = s_{i_1} \rightarrow s_{i_2} \rightarrow \dots \rightarrow s_{i_k}$ in the prefix graph define

$$s(C) = \text{pref}(s_{i_1}, s_{i_2})\text{pref}(s_{i_2}, s_{i_3}), \dots, \text{pref}(s_{i_k}, s_{i_1}).$$

Then $|s(C)|$ is the length of C and $s(C)$ essentially reads the prefixes along the cycle. The strings $s(C)$ for $C \in \mathcal{C}_{\min}$ have very interesting properties.

Lemma 2.5 (Claims 3 and 5 in Blum et al. [4]). *The strings $s(C)$ are all primitive with $|s(C)| \geq 2$ and are all non-equivalent.*

Let $w(C)$ be the nice rotation of $s(C)$ for every $C \in \mathcal{C}_{\min}$. As we already mentioned, the representative strings $R(C)$ defined as earlier are unnecessary long. This can be used to prove the following.

Lemma 2.6 (Special case of Lemma 5.1 in Breslauer et al. [5]). *One can define the representative $R(C)$ for a cycle $s_{i_1} \rightarrow \dots \rightarrow s_{i_k}$ so that:*

- $R(C)$ is a substring of $\langle s_{i_j}, s_{i_{j+1}}, \dots, s_{i_k}, s_{i_1}, \dots, s_{i_j} \rangle$ for some j (in particular $OPT(R) \leq 2OPT(S)$ still holds),
- $R(C)$ is a $w(C)$ -word.

Finally, we need to show that the strings $R(C)$ do not overlap too much. The lemma below is stated in a slightly more general fashion so that it can be used more easily later on.

Lemma 2.7 (Implicit in the proof of Lemma 3.3 in Breslauer et al. [5]). *Let w_1 and w_2 be non-equivalent nice words and let x_i be a w_i -word for $i = 1, 2$. Also let $\alpha_i = \alpha(w_i)$ and $l_i = |w_i|$. Then $|\text{ov}(x_1, x_2)| < l_1 + \alpha_2$.*

Proof. Assume for a contradiction that $|\text{ov}(x_1, x_2)| \geq l_1 + \alpha_2$. Consider the string $z = \text{ov}(x_1, x_2)[l_1 + 1, l_1 + \alpha_2]$. We have $z = \text{ov}(x_1, x_2)[1, \alpha_2] = w_2[1, \alpha_2]$, so we need to have $l_1 = kl_2$ for some k because of Lemma 2.2, which is impossible.

To see why, notice that if $l_1 = kl_2$ and $|\text{ov}(x_1, x_2)| \geq l_1$, then either w_1 and w_2 are equivalent (if $k = 1$) or w_1 is nonprimitive (if $k > 1$). \square

Theorem 2.8 (Breslauer et al. [5]). *Given c -approximation for MAX-ATSP-PATH, one can approximate SHORTEST-SUPERSTRING with approximation factor of $3\frac{1}{2} - 1\frac{1}{2}c$.*

Proof. Consider the string $s = \langle R(C_1), \dots, R(C_k) \rangle$ that is the optimal solution for R . Let R_{OV} be the total overlap of this string. Then by applying Lemma 2.7 to every pair of consecutive strings we get

$$R_{OV} \leq \sum_{i=1}^{k-1} (|w(C_i)| + \alpha(w(C_{i+1}))) \leq \frac{3}{2} \sum_{i=1}^k |w(C_i)| \leq \frac{3}{2} OPT(S).$$

A c -approximation algorithm for MAX-ATSP-PATH can be used to obtain a solution with total overlap of cR_{OV} . The length of the resulting SHORTEST-SUPERSTRING solution is therefore at most

$$OPT(R) + \frac{3}{2}(1 - c)OPT(S) \leq 2OPT(S) + \frac{3}{2}(1 - c)OPT(S) = \left(3\frac{1}{2} - 1\frac{1}{2}c\right) OPT(S).$$

\square

Since $\frac{2}{3}$ -approximation algorithms for MAX-ATSP-PATH are known we obtain the following

Corollary 2.9 (Kaplan et al. [10], also Paluch et al. [16]). *There exists a $2\frac{1}{2}$ -approximation algorithm for SHORTEST-SUPERSTRING.*

3 The Algorithm

In this section we give the new approximation algorithm and bound its approximation factor.

Description The algorithm we are going to analyse is very simple. It returns a solution S_0 which is the better of the following two solutions S_1, S_2 :

- S_1 is obtained by using any algorithm that reduces SHORTEST-SUPERSTRING to MAX-ATSP-PATH (e.g. one due to Kaplan et al. [10] or Paluch et al. [16]),
- S_2 is also obtained by reducing to MAX-ATSP-PATH, but this time we get the final solution by computing the maximum weight cycle cover in the overlap graph of R and dropping the lightest edge from every cycle.

Analysis For any cycle C in the overlap graph let O_C be the total overlap of C , i.e. sum of the weights of its edges. Let M_C be the minimum weight of an edge of C . Also, let L_C be the sum of the periods of the strings in C , which is equal to the total length of the corresponding cycles in \mathcal{C}_{\min} .

Let $|R|$ be the total length of the representative strings in R and let \mathcal{C} be the maximum weight cycle cover in the overlap graph of R . Note that $\sum_{C \in \mathcal{C}} L_C = w(\mathcal{C}_{\min})$. Moreover, let $O_{\mathcal{C}} = \sum_{C \in \mathcal{C}} O_C$, let $M_{\mathcal{C}} = \sum_{C \in \mathcal{C}} M_C$. Finally, let c be the best known approximation ratio for MAX-ATSP-PATH.

Lemma 3.1. $|S_1| \leq OPT(R) + (1 - c)O_{\mathcal{C}}$ and $|S_2| \leq OPT(R) + M_{\mathcal{C}}$.

Proof. We have $|R| - O_{\mathcal{C}} \leq OPT(R)$ and $|S_2| \leq |R| - O_{\mathcal{C}} + M_{\mathcal{C}}$ which proves the second part.

For the first, note that since $O_{\mathcal{C}} \geq |R| - OPT(R)$, any algorithm that approximates $|R| - OPT(R)$ with factor c gives a solution of length at most $|S_1| \leq |R| - c(|R| - OPT(R)) = OPT(R) + (1 - c)(|R| - OPT(R)) \leq OPT(R) + (1 - c)O_{\mathcal{C}}$. \square

The main technical ingredient of this paper is the following theorem (we show in Section 6 that it is essentially tight).

Theorem 3.2 (Main Theorem (local version)). *For any cycle C in the overlap graph, we have*

$$2M_C + 7O_C \leq 11L_C.$$

By summing over all cycles of \mathcal{C} we obtain the following.

Theorem 3.3 (Main Theorem (global version)).

$$2M_{\mathcal{C}} + 7O_{\mathcal{C}} \leq 11w(\mathcal{C}_{\min}) \leq 11OPT(S).$$

The next two sections are devoted to the proof of Theorem 3.2. For now let us see what it implies.

Corollary 3.4. *SHORTEST-SUPERSTRING can be approximated with factor $\left(2 + \frac{11(1-c)}{9-2c}\right)$. In particular for $c = \frac{2}{3}$ we get $2\frac{11}{23}$ -approximation.*

Proof. It follows from Lemma 3.1 and Lemma 2.4 that

$$|S_0| \leq 2OPT(S) + \min(M_C, (1-c)O_C).$$

We can bound the second term as follows:

$$\min(M_C, (1-c)O_C) \leq \frac{2-2c}{9-2c}M_C + \frac{7}{9-2c}(1-c)O_C \leq \frac{11(1-c)}{9-2c}OPT(S).$$

For $c = \frac{2}{3}$ we obtain $\frac{11 \cdot \frac{1}{3}}{9-2 \cdot \frac{2}{3}} = \frac{11}{23}$ and so $|S_0| \leq 2\frac{11}{23}OPT(S)$. \square

4 The General Bounds

In this section we present and prove the bounds on overlaps of strings. We consider a set of non-equivalent nice strings w_1, \dots, w_k , and for each $i = 1, \dots, k$ a w_i -string x_i . We use l_i to denote $|w_i|$, and α_i to denote $\alpha(w_i)$. Moreover, for each $i \neq j$, $i, j \in \{1, \dots, k\}$ we define $ov_{ij} = ov(x_i, x_j)$ and $o_{ij} = |ov_{ij}|$. Finally, let w_{ij} be the rotation of w_i that matches $ov(x_i, x_j)$ from the left. If there is more than one such rotation (which might happen if $o_{ij} < l_i$), choose any such rotation.

By Lemma 2.7 we have $o_{12} \leq l_1 + \frac{1}{2}l_2$. The main theme of this section is characterizing situations in which this inequality is in some way non-tight. The underlying idea in most (but not all) of these results is the following: We show that if o_{12} is actually close to its upper-bound, then the set of possible starting positions of the maximal/minimal rotation of w_1 is strongly limited, which in turn leads to an upper-bound on α_1 . This can be used to upper-bound other overlaps using Lemma 2.7.

We start with another lemma from the work of Breslauer et al. [5].

Lemma 4.1 (Implicit in the proof of Lemma 3.3 of Breslauer et al. [5]). *If $l_1 \leq l_2$ then $o_{12} < l_2$. For general l_1, l_2 we have $o_{12} < kl_2$ whenever $l_1 \leq kl_2$.*

Proof. Assume for a contradiction that $l_1 \leq kl_2$ and $o_{12} \geq kl_2$. Also, w.l.o.g. assume that k is the smallest integer such that $l_1 \leq kl_2$.

Similarly as in the proof of Lemma 2.7 we cannot have $l_1 = kl_2$, and we also cannot have $l_1 = (k-1)l_2$ for the same reasons. Therefore $(k-1)l_2 < l_1 < kl_2$.

Consider now the string $ov_{12}[l_1 + 1, kl_2] = ov_{12}[1, kl_2 - l_1]$. This string is a non-trivial suffix of w_2 , and also a prefix of w_2 , a contradiction with w_2 being nice and Lemma 2.3. \square

The next two lemmas demonstrate that o_{12} getting close to $l_1 + \frac{1}{2}l_2$ implies an upper-bound on the value of α_1 . While Lemma 4.3 gives this bound explicitly, Lemma 4.2 describes it in terms of constraints on the starting positions of maximal and minimal rotations of w_1 .

Lemma 4.2. *Let $l_1 \geq l_2$, $o_{12} \geq l_2$ and let w_2 be its maximal rotation, then:*

- $i_{\max}(w_{12}) = 1$, or $i_{\max}(w_{12}) = l_2 \lfloor \frac{o_{12}-1}{l_2} \rfloor + 1$ or $i_{\max}(w_{12}) > \max(l_2 \lfloor \frac{o_{12}-1}{l_2} \rfloor + 1, o_{12} - \alpha_2 + 1)$.

- $i_{\min}(w_{12}) = \alpha_2 + 1$, or $i_{\min}(w_{12}) = l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$, or $i_{\min}(w_{12}) > \max(l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1, o_{12} - (l_2 - \alpha_2) + 1)$,

Moreover, if $o_{12} \geq l_1$, then $i_{\max}(w_{12}) \neq 1$ and $i_{\min}(w_{12}) = \alpha_2 + 1$.

Remark. Several of the lemmas appearing in the remainder of this section assume that one of the strings involved is its maximal rotation. In all cases a symmetrical statement is true as well, in which the roles of minimal and maximal rotations of all strings involved are reversed. We omit the corresponding statements in all these lemmas.

Proof (of Lemma 4.2). Let us start with claims concerning $i_{\max}(w_{12})$. Since we assume $o_{12} \geq l_2$, we know that w_{12} contains $p_{\max}(w_2)$. Therefore $(w_{12})_{\max}[1, \alpha_2] \succeq p_{\max}(w_2)$. We now consider two cases, depending on whether or not $i_{\max}(w_{12}) \leq o_{12} - \alpha_2 + 1$.

Case 1: If $i_{\max}(w_{12}) \leq o_{12} - \alpha_2 + 1$, then $i_{\max}(w_{12}) + \alpha_2 - 1 \leq o_{12}$ and consequently $(w_{12})_{\max}[1, \alpha_2] \preceq p_{\max}(w_2)$. By the previous observation this is in fact an equality, and by Lemma 2.2 $i_{\max}(w_2) = kl_2 + 1$ for some natural k , i.e. the maximal rotation of w_{12} is aligned with the starting position of some occurrence of w_2 in o_{12} . The two positions that appear in the statement of the lemma: $\alpha_2 + 1$ and $l_2 \lfloor \frac{o_{12} - 1}{l_2} \rfloor + 1$, are the starting positions of the leftmost and the rightmost occurrences, respectively. We will show that $i_{\max}(w_{12})$ has to be equal to one of them.

Suppose that this is not the case. This means that we have $i_{\max}(w_{12}) = kl_2 + 1$ and both $(k - 1)l_2 + 1$ and $(k + 1)l_2 + 1$ are in $[1, \dots, o_{12}]$. Note that by Lemma 4.1 we then also have $(k + 1)l_2 \leq l_1$, and in fact $(k + 1)l_2 < l_1$ since otherwise l_1 would not be primitive. Therefore $(k + 1)l_2 + 1 \leq l_1$. Consider the rotations r_1, r_2, r_3 of w_{12} starting at positions $(k - 1)l_2 + 1$, $kl_2 + 1$ and $(k + 1)l_2 + 1$, respectively. We have $r_1 = w_2 w_2 w$, $r_2 = w_2 w w_2$ and $r_3 = w w_2 w_2$ for some string w . Since all rotations of a primitive string are different, we have $ww_2 \neq w_2 w$. If $ww_2 \succ w_2 w$, then r_3 is the largest of the three rotations. If, on the other hand, $ww_2 \prec w_2 w$, then r_1 is the largest one. Therefore $i_{\max}(w_{12}) \neq kl_2 + 1$, a contradiction.

Case 2: We are left with the case where $i_{\max}(w_{12}) > o_{12} - \alpha_2 + 1$, and we can also assume that we do not have $i_{\max}(w_{12}) = l_2 \lfloor \frac{o_{12} - 1}{l_2} \rfloor + 1$ since then the lemma clearly holds.

We need to prove that $i_{\max}(w_{12}) > l_2 \lfloor \frac{o_{12} - 1}{l_2} \rfloor + 1$. If that was not the case, then $(k - 1)l_2 + 1 < i_{\max}(w_{12}) \leq kl_2 < o_{12}$ for $k = \lfloor \frac{o_{12} - 1}{l_2} \rfloor$. Then $w_{12}[i_{\max}(w_{12}), kl_2]$ is a non-trivial suffix of w_2 , and it is also a prefix of w_2 by maximality of the rotation of w_{12} starting at $i_{\max}(w_{12})$. But that is a contradiction with the fact that w_2 is unbordered by Lemma 2.3. This ends the proof of the bounds for $i_{\max}(w_{12})$.

One final claim we need to show concerning $i_{\max}(w_{12})$ is that if $o_{12} \geq l_1$, then $i_{\max}(w_{12}) \neq 1$. Since for $o_{12} \geq l_1$ we have $l_1 > l_2$, there are at least two positions of the form $kl_2 + 1$ within w_{12} . Consider rotations r_1 and r_2 of w_{12} starting at two consecutive such positions $kl_2 + 1$ and $(k + 1)l_2 + 1$. We will prove that $r_1 \prec r_2$, which implies our claim. We have $r_1 = w_2 w w_2^k$ and $r_2 = w w_2^{k+1}$ for some w which is a prefix of w_2^∞ . In particular, we have $r_1 = v w w_2^k$ for some v which is a rotation of w_2 . Since $r_1 \neq r_2$ by Lemma 2.2, and w_2 is its maximal rotation, we conclude that $r_1 \prec r_2$.

Let us now prove the claims concerning $i_{\min}(w_{12})$. Similarly to the case of $i_{\max}(w_{12})$ we can argue that either $i_{\min}(w_{12}) > o_{12} - (l_2 - \alpha_2) + 1$ or we have $i_{\min}(w_{12}) = kl_2 + \alpha_2 + 1$ for some k .

This time it will be more convenient to start with the case of $o_{12} \geq l_1$. Among rotations starting at positions of the form $kl_2 + \alpha_2 + 1$, the one starting at $\alpha_2 + 1$ is minimal in this case, and the proof is almost identical to the one we just presented for $i_{\max}(w_{12})$.

So we only need to exclude the case where $o_{12} \geq l_1$ and $i_{\min}(w_{12}) > o_{12} - (l_2 - \alpha_2) + 1$. If that happened, then we would have $(w_{12})_{\min}[1, l_2 - \alpha_2] = w_{12}[i_{\min}(w_{12}), l_1]w$, where w is a non-empty prefix of w_2 . Call this string r_1 and let $r_2 = w_{12}[\alpha_2 + 1, l_2]$. We claim that $r_2 \prec r_1$, which is a contradiction with minimality of r_1 . To see that $r_2 \prec r_1$, note that $w_{12}[\alpha_2 + 1, \alpha_2 + l_1 - i_{\min}(w_{12})] \preceq w_{12}[i_{\min}(w_{12}), l_1]$ by the definition of α_2 . Moreover $w_{12}[l_2 - |w| + 1, l_2] \preceq w$ because w is a prefix of $(w_2)_{\max}$. But we cannot have an equality here, since then w would also be a suffix of $(w_2)_{\max}$, a contradiction with Lemma 2.3.

Let us now prove the main claims concerning $i_{\min}(w_{12})$. Again, we consider two cases.

Case 1: If $i_{\min} \leq o_{12} - (l_2 - \alpha_2) + 1$ and consequently $i_{\min}(w_{12})$ is of the form $kl_2 + \alpha_2 + 1$, then we can show that we have either $i_{\min}(w_{12}) = \alpha_2 + 1$ or $i_{\min}(w_{12}) = l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$. The proof is almost identical to the one we provided for $i_{\max}(w_{12})$. The only step that does not directly translate, is that the rightmost position of the form $kl_2 + \alpha_2 + 1$ within ov_{12} is also within w_{12} . Luckily, we already considered the case of $o_{12} \geq l_1$.

Case 2: If $i_{\min}(w_{12}) > o_{12} - (l_2 - \alpha_2) + 1$, then we can also assume that we do not have $i_{\min}(w_{12}) = l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$. We need to show that $i_{\min}(w_{12}) > l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$. Again, the proof is almost identical to the one for $i_{\max}(w_{12})$. \square

Lemma 4.3. *Let $l_1, o_{12} \geq l_2$ and let w_2 be its maximal rotation. Then we always have $\alpha_1 \leq l_2 + (l_1 + \alpha_2 - o_{12})$ and moreover:*

1. *either $\alpha_1 \leq l_1 + l_2 - o_{12}$, or*
2. *the maximal rotation of w_{12} starts at the rightmost position of the form $kl_2 + 1$, or the minimal rotation of w_{12} starts at the rightmost position of the form $kl_2 + \alpha_2 + 1$.*

Proof. Since we assume $o_{12} \geq l_2$, Lemma 4.2 describes all possibilities for $i_{\min}(w_{12})$ and $i_{\max}(w_{12})$. The rest is simple case analysis.

If either $i_{\max}(w_{12}) = l_2 \lfloor \frac{o_{12} - 1}{l_2} \rfloor + 1$ or $i_{\min}(w_{12}) = l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$ (i.e. the second alternative in the statement of the lemma holds) then $i_{\min}(w_{12})$ is either at most $\alpha_2 + 1$ or at least $o_{12} - l_2 + 1$ by Lemma 4.2, and the same bounds hold for $i_{\max}(w_{12})$. “Wrapping around” the end of w_{12} , they both land in an interval of length $l_1 - (o_{12} - l_2) + \alpha_2 = l_2 + (l_1 + \alpha_2 - o_{12})$ and hence this quantity is also an upper bound on α_1 .

If neither $i_{\max}(w_{12}) = l_2 \lfloor \frac{o_{12} - 1}{l_2} \rfloor + 1$ nor $i_{\min}(w_{12}) = l_2 \lfloor \frac{o_{12} - \alpha_2 - 1}{l_2} \rfloor + \alpha_2 + 1$, then Lemma 4.2 gives even stronger bounds. We have $i_{\min}(w_{12}) \leq \alpha_2 + 1$ or $i_{\min}(w_{12}) > o_{12} - (l_2 - \alpha_2) + 1$ and the same bounds (in fact stronger) hold for $i_{\max}(w_{12})$. Repeating the previous argument we get

$$\alpha_1 \leq l_1 - (o_{12} - (l_2 - \alpha_2)) + \alpha_2 = l_1 + l_2 - o_{12}.$$

\square

The next two lemmas state some of the consequences of Lemmas 4.2 and 4.3, that are particularly easy to use.

Lemma 4.4. *If $l_1 \geq l_2$ then:*

1. $o_{12} + \alpha_1 \leq l_1 + l_2$ for $l_1 < 2l_2$,
2. $o_{12} + \alpha_1 \leq 2l_1 - l_2 = l_1 + l_2 + (l_1 - 2l_2)$ for $2l_2 \leq l_1 < \frac{5}{2}l_2$,
3. $o_{12} + \alpha_1 \leq l_1 + l_2 + \alpha_2$.

Proof. In the proof, we assume w.l.o.g. that w_2 is its maximal rotation.

Let us first consider the case of $l_1 < 2l_2$. If $o_{12} < l_1$ then we get the claim, since $\alpha_1 \leq \frac{1}{2}l_1 \leq l_2$. On the other hand, if $o_{12} \geq l_1$ (note that in this case $l_1 > l_2$), then by Lemma 4.2 we have $i_{\min}(w_{12}) = \alpha_2 + 1$ and $i_{\max}(w_{12}) \in \{l_2 + 1\} \cup (o_{12} - \alpha_2 + 1, \dots, l_1]$. Therefore, we either have $\alpha_1 \leq l_2 - \alpha_2$ or $\alpha_1 \leq (l_1 - o_{12} + \alpha_2) + \alpha_2$ and in both cases it is easy to verify that our claim is true.

To prove the second inequality, we consider three cases:

Case 1: If $o_{12} < l_1$ then

$$o_{12} + \alpha_1 \leq l_1 + \alpha_1 \leq \frac{3}{2}l_1 \leq 2l_1 - l_2.$$

Case 2: If the first alternative in Lemma 4.3 holds, i.e. $\alpha_1 \leq l_1 + l_2 - o_{12}$ then

$$o_{12} + \alpha_1 \leq l_1 + l_2 \leq 2l_1 - l_2.$$

Case 3: We are left with the case where the second alternative of Lemma 4.3 holds. Since $o_{12} \geq l_1$ and so $i_{\min}(w_{12}) = \alpha_2 + 1$, this means that $i_{\max}(w_{12}) = 2l_2 + 1$. It follows that $\alpha_1 \leq (l_1 - 2l_2) + \alpha_2$ and so

$$o_{12} + \alpha_1 \leq (l_1 + \alpha_2) + (l_1 - 2l_2) + \alpha_2 \leq 2l_1 - l_2.$$

The third inequality of Lemma 4.4 follows immediately from the inequality $\alpha_1 \leq l_2 + (l_1 + \alpha_2 - o_{12})$ in the first part of Lemma 4.3. \square

Let $\Delta o_{ij} = (l_i + \frac{1}{2}l_j) - o_{ij}$ and $\Delta \alpha_i = \frac{1}{2}l_i - \alpha_i$. These basically measure how much smaller o_{ij} and α_i are from their maximum values.

Corollary 4.5.

1. $\Delta o_{12} + \Delta \alpha_1 \geq \frac{1}{2}(l_1 - l_2)$ if $l_1 < 2l_2$,
2. $\Delta o_{12} + \Delta \alpha_1 \geq \frac{1}{4}(l_1 - l_2)$ if $l_1 \geq 3l_2$, and
3. $\Delta o_{12} + \Delta \alpha_1 \geq \frac{1}{6}(l_1 - l_2)$.

Proof. For the first inequality, we have by Lemma 4.4

$$\Delta o_{12} + \Delta \alpha_1 = \frac{3}{2}l_1 + \frac{1}{2}l_2 - o_{12} - \alpha_1 \geq \frac{3}{2}l_1 + \frac{1}{2}l_2 - l_1 - l_2 = \frac{1}{2}(l_1 - l_2).$$

For the second inequality, we have by Lemma 4.4

$$\Delta o_{12} + \Delta \alpha_1 \geq \frac{3}{2}l_1 + \frac{1}{2}l_2 - (l_1 + l_2 + \alpha_2) = \frac{1}{2}l_1 - \frac{1}{2}l_2 - \alpha_2 \geq \frac{1}{2}l_1 - l_2 \geq \frac{1}{4}l_1 + \frac{3}{4}l_2 - l_2 = \frac{1}{4}(l_1 - l_2).$$

Clearly, we only need to prove the third inequality for $2l_2 \leq l_1 < 3l_2$. We consider two cases:

Case 1: If $2l_2 \leq l_1 \leq \frac{5}{2}l_2$ then

$$\Delta o_{12} + \Delta \alpha_1 \geq \frac{3}{2}l_1 + \frac{1}{2}l_2 - (2l_1 - l_2) = \frac{3}{2}l_2 - \frac{1}{2}l_1 = \frac{1}{6}(l_1 - l_2) + \frac{10}{6}l_2 - \frac{4}{6}l_1 \geq \frac{1}{6}(l_1 - l_2).$$

Case 2: If $\frac{5}{2}l_2 \leq l_1 < 3l_2$ then we have

$$\Delta o_{12} + \Delta \alpha_1 \geq \frac{3}{2}l_1 + \frac{1}{2}l_2 - (l_1 + l_2 + \alpha_2) \geq \frac{1}{2}l_1 - l_2 \geq \frac{1}{6}l_1 + \frac{5}{2} \cdot \frac{1}{3}l_2 - l_2 = \frac{1}{6}(l_1 - l_2).$$

\square

The last lemma is not used in the proof of the main theorem. Nevertheless, we decided to include it in this section, as we believe it might turn out useful in future developments.

Lemma 4.6. *Let $l_1 \leq l_2$ and let w_2 be its maximal rotation. If $o_{12} \geq l_1 + \alpha_2 - \alpha_1$, then $\alpha_1 \leq |\alpha_2 - kl_1|$ for all positive integers k .*

Remark. *The most important consequence of the above lemma is that if $l_2 \sim 2l_1$, then we cannot have $\alpha_1 \sim \frac{1}{2}l_1$, $\alpha_2 \sim \frac{1}{2}l_2$ and $o_{12} \sim l_1 + \alpha_2$ all happening at the same time.*

Proof. We have $o_{12} \geq l_1 - \alpha_1 \geq |p_{\max}(w_{12})|$, so $w_2[1, |p_{\max}(w_{12})|] = (w_2)_{\max}[1, |p_{\max}(w_{12})|]$ is a substring of ov_{12} , and in particular $p_{\max}(w_{12}) \succeq (w_2)_{\max}[1, |p_{\max}(w_{12})|]$. Therefore, if $p_{\max}(w_{12})$ is contained in ov_{12} we need to have $p_{\max}(w_{12}) = w_{12}[1, |p_{\max}(w_{12})|]$ and $i_{\max}(w_{12}) = 1$ by Lemma 2.2. If, on the other hand, $p_{\max}(w_{12})$ is not contained in ov_{12} , then we have $i_{\max}(w_{12}) > o_{12} - (l_1 - \alpha_1) + 1 \geq \alpha_2 + 1$ and so $i_{\max}(w_{12}) \in [\alpha_2 + 1, \dots, l_1]$.

Similarly we can see that ov_{12} contains $w_2[\alpha_2 + 1, \alpha_2 + |p_{\min}(w_{12})|]$, and so if $p_{\min}(w_{12})$ is contained in ov_{12} , we have $p_{\min}(w_{12}) = x_2[\alpha_2 + 1, \alpha_2 + |p_{\min}(w_{12})|]$, and by Lemma 2.2 $i_{\min}(w_{12}) = (\alpha_2 + 1) \bmod l_1$. Otherwise, we have $i_{\min}(w_{12}) \in [\alpha_2 + 1, \dots, l_1]$.

It is easy to verify that in all cases for $i_{\max}(w_{12})$ and $i_{\min}(w_{12})$ we get $\alpha_1 \leq |\alpha_2 - kl_1|$ for all positive integers k . \square

5 The Proof of the Main Theorem

In this section we present the proof of Theorem 3.2. We first introduce some additional definitions and technical lemmas, designed specifically for this proof, in Subsection 5.1. Since the proof itself is a rather long and detailed case analysis, in Subsection 5.2 we present a simple proof of a weaker version of Theorem 3.2. This weaker statement still gives an approximation ratio below $2\frac{1}{2}$. The proof of Theorem 3.2 follows, for easier reading split into a subsection covering some basic observations and four subsections corresponding to different cycle lengths.

5.1 Preliminaries

We keep the notation from previous chapters. In particular, for a cycle $C = x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_k \rightarrow x_1$, we are interested in bounding $M = M_C = \min\{o_{ij} : (x_i, x_j) \in C\}$ and $O = O_C = \sum_{(x_i, x_j) \in C} o_{ij}$ in terms of $L = L_C = \sum_{i=1}^k l_i$. Recall also, that $\Delta o_{ij} = (l_i + \frac{1}{2}l_j) - o_{ij}$ and $\Delta \alpha_i = \frac{1}{2}l_i - \alpha_i$. Let $\Delta O = \sum_{(x_i, x_j) \in C} \Delta o_{ij} = \frac{3}{2}L - O$.

We now introduce a couple more definitions. We call an edge $x_i \rightarrow x_j$ a *down-edge* if $l_i \geq l_j$ and we call it an *up-edge* otherwise. We denote the sets of down-edges and up-edges of C by C_d and C_u respectively. A down-edge $x_i \rightarrow x_j$ is *steep* if $l_i \geq 2l_j$, otherwise it is *flat*. Similarly an up-edge $x_i \rightarrow x_j$ is *steep* if $l_i \leq \frac{1}{2}l_j$, and *flat* otherwise.

Finally let l_{\min} and $l_{\max} = l_1$ be the smallest and the largest among l_1, \dots, l_k breaking ties arbitrarily.

Lemma 5.1. *For any up-edge $x_i \rightarrow x_j$ we have*

$$\Delta o_{ij} \geq l_i - \frac{1}{2}l_j \geq l_{\min} - \frac{1}{2}l_{\max}.$$

Proof. The second inequality is obvious.

As for the first, there is nothing to prove for steep up-edges since then $l_i - \frac{1}{2}l_j \leq 0$. For flat up-edges we have

$$\Delta o_{ij} \geq \left(l_i + \frac{1}{2}l_j\right) - l_j = l_i - \frac{1}{2}l_j,$$

by Lemma 4.1. □

Lemma 5.2. *For any cycle C we have*

$$\Delta O \geq \frac{1}{12} \sum_{(x_i, x_j) \in C_d} (l_i - l_j) \geq \frac{1}{12} (l_{\max} - l_{\min}).$$

The constant can be improved to $\frac{1}{8}$ if there are no two consecutive steep down-edges in C , and to $\frac{1}{4}$, if there are no steep down-edges in C .

Proof. Let $x_i \rightarrow x_j$ be a down-edge, and let $x_l \rightarrow x_i$ be the edge preceding it on C . Then we get from Corollary 4.5:

$$\Delta o_{li} + \Delta o_{ij} \geq \Delta \alpha_i + \Delta o_{ij} \geq \frac{1}{6} (l_i - l_j) \quad (1)$$

The right-hand side of the sum of inequality (1) over all down-edges is upper-bounded by $2\Delta O$ and the claim follows.

If there are no steep down-edges in C , then this reasoning can be repeated using the sharper bound in Corollary 4.5.

Finally, if there are no two consecutive steep down-edges in C , then let C_s be the set of steep down-edges and consider the sum of inequality (1) over C_s . Since steep down-edges are nonconsecutive, the right-hand side of this inequality is upperbounded by ΔO , and so

$$\Delta O \geq \frac{1}{6} \sum_{(x_i, x_j) \in C_s} (l_i - l_j).$$

We can also slightly improve the first part of the proof by using a stronger bound for flat edges to obtain:

$$2\Delta O \geq \frac{1}{6} \sum_{(x_i, x_j) \in C_s} (l_i - l_j) + \frac{1}{2} \sum_{(x_i, x_j) \in C_d \setminus C_s} (l_i - l_j).$$

Adding twice the first inequality to the second one, we get

$$4\Delta O \geq \frac{1}{2} \sum_{(x_i, x_j) \in C_s} (l_i - l_j) + \frac{1}{2} \sum_{(x_i, x_j) \in C_d \setminus C_s} (l_i - l_j),$$

and the claim follows. □

Lemma 5.3. *If $l_{\min} > \frac{1}{2}l_{\max}$ then*

$$\Delta O \geq \frac{1}{4}l_{\min}.$$

Proof. Note that in the proof of Lemma 5.2 we actually have

$$2\Delta O \geq \frac{1}{6} \sum_{(x_i, x_j) \in C_s} (l_i - l_j) + \frac{1}{2} \sum_{(x_i, x_j) \in C_d \setminus C_s} (l_i - l_j) + \sum_{(x_i, x_j) \in C_u} \Delta o_{ij}.$$

If $l_{\min} > \frac{1}{2}l_{\max}$, then since all edges are flat and there is at least one down-edge (we excluded the case of all l_i equal) we obtain

$$2\Delta O \geq \frac{1}{2} \sum_{(x_i, x_j) \in C_d} (l_i - l_j) + \left(l_{\min} - \frac{1}{2}l_{\max}\right) \geq \frac{1}{2}(l_{\max} - l_{\min}) + \left(l_{\min} - \frac{1}{2}l_{\max}\right) = \frac{1}{2}l_{\min},$$

by Lemma 5.1 and the claim follows. \square

5.2 Proof of a Weaker Version of the Main Theorem

In this subsection we present a weaker version of the Theorem 3.2, which is relatively easy to prove, and still leads to approximation factor smaller than $2\frac{1}{2}$.

Theorem 5.4 (Main Theorem, weak local version). *For any cycle C in the overlap graph of R we have*

$$M_C + 24O_C \leq 36\frac{1}{4}L_C.$$

Remark. *Note that the emphasis here is on simplicity, and the proof below can easily be improved in many ways.*

Proof. We first prove that we always have $\Delta O \geq \frac{1}{24k}L$, where k is the length of the cycle. We consider two cases:

Case 1: If $l_{\max} < 2l_{\min}$ then by Lemma 5.3 we have $\Delta O \geq \frac{1}{4}l_{\min}$ and so

$$\Delta O \geq \frac{1}{4}l_{\min} \geq \frac{1}{4(2k-1)}(2k-1)l_{\min} \geq \frac{1}{4(2k-1)}(l_{\min} + (k-1)l_{\max}) \geq \frac{1}{4(2k-1)}L \geq \frac{1}{24k}L.$$

Case 2: If $l_{\max} \geq 2l_{\min}$ then we have by Lemma 5.2

$$\begin{aligned} \Delta O &\geq \frac{1}{12}(l_{\max} - l_{\min}) \geq \frac{1}{12(2k-1)}((2k-1)l_{\max} - (2k-1)l_{\min}) \geq \\ &\geq \frac{1}{12(2k-1)}((k-1)l_{\max} + l_{\min}) \geq \frac{1}{12(2k-1)}L \geq \frac{1}{24k}L. \end{aligned}$$

From the inequality we have just proved we get

$$O \leq \left(\frac{3}{2} - \frac{1}{24k}\right)L.$$

We also always have

$$M \leq \frac{1}{k} \cdot O \leq \frac{3}{2k}L.$$

Joining these gives

$$M + 24O \leq \left(\frac{3}{2k} + 36 - \frac{1}{k}\right)L = \left(36 + \frac{1}{2k}\right)L \leq 36\frac{1}{4}L.$$

\square

Corollary 5.5. *There exists a $2\frac{145}{292}$ -approximation algorithm for SHORTEST-SUPERSTRING.*

Proof. Similarly as in the proof of Corollary 3.4, we get from Lemma 3.1 and Lemma 2.4 that

$$|S_0| \leq 2OPT(S) + \min\left(M, \left(1 - \frac{2}{3}\right)O\right).$$

We can bound the second term as follows:

$$\min\left(M, \frac{1}{3}O\right) \leq \left(\frac{1}{73}M + \frac{72}{73} \cdot \frac{1}{3}O\right) \leq \frac{36\frac{1}{4}L}{73} = \frac{145}{292}L \leq \frac{145}{292}OPT(S),$$

and so $|S_0| \leq 2\frac{145}{292}OPT(S)$. \square

5.3 The Proof of Theorem 3.2 – Basic Observations

Let us recall Theorem 3.2.

Theorem 3.2 (restated). *For every cycle C in the overlap graph of R , we have*

$$2M_C + 7O_C \leq 11L_C.$$

We can easily get rid of the following special case, which will make some reasonings easier later on.

Lemma 5.6. *If all l_i are equal for a cycle C then the claim of Theorem 3.2 holds.*

Proof. Since two non-equivalent strings of equal length l cannot have an overlap of length l or greater, it follows that in this case $\gamma \leq 1$. Therefore $\beta \leq \frac{1}{2}\gamma \leq \frac{1}{2}$ and $2\beta + 7\gamma \leq 8$, a much stronger bound than needed. \square

In the remainder of this section we assume that not all l_i are equal.

Lemma 5.7. *Either of the following statements imply the claim of Theorem 3.2 for a k -cycle C :*

- $2M - 7\Delta O \leq \frac{1}{2}$,
- $\Delta O \geq \frac{6-k}{2(7k+2)}L$.

Before proving the above lemma, let us note its particularly useful consequences:

Corollary 5.8. *Let C be a k -cycle, then the claim of Theorem 3.2 for C*

- *is implied by $\Delta O \geq \frac{1}{30}L$ if $k = 4$,*
- *is implied by $\Delta O \geq \frac{1}{74}L$ if $k = 5$,*
- *holds if $k \geq 6$.*

Proof (of Lemma 5.7). For the first part we have

$$2M + 7O = 2M + 7\left(\frac{3}{2}L - \Delta O\right) = 10\frac{1}{2}L + (2M - 7\Delta O),$$

and the claim follows.

For the second part, note that $M \leq \frac{1}{k}O$. Therefore if we have $\Delta O \geq \frac{6-k}{2(7k+2)}L$, then

$$2M + 7O \leq \frac{2+7k}{k}O \leq \frac{2+7k}{k}\left(\frac{3}{2}L - \frac{6-k}{2(7k+2)}L\right) = \frac{2+7k}{k} \cdot \frac{22k}{2(7k+2)}L = 11L.$$

\square

5.4 The Proof of Theorem 3.2 for 5-cycles

The remainder of the proof is divided into four parts, one for each cycle length in $\{2, 3, 4, 5\}$. Although there are similarities between these parts, they are mostly independent. For easier reading, we put each part in a separate subsection.

Lemma 5.9. *If C is 5-cycle, then $2M + 7O \leq 11L$.*

Proof. We consider three cases. In all three we prove that $\Delta O \geq \frac{1}{74}L$ and the claim follows from Corollary 5.8.

Case 1: If $l_{\min} > \frac{1}{2}l_{\max}$ we have by Lemma 5.3 that $\Delta O \geq \frac{1}{4}l_{\min}$ and so

$$\Delta O \geq \frac{1}{36}(9l_{\min}) \geq \frac{1}{36}(l_{\min} + 4l_{\max}) \geq \frac{1}{36}L.$$

Case 2: If $\frac{1}{4}l_{\max} < l_{\min} \leq \frac{1}{2}l_{\max}$, then we cannot have two consecutive steep down-edges, and so by Lemma 5.2 we have

$$\Delta O \geq \frac{1}{8}(l_{\max} - l_{\min}) \geq \frac{1}{72}(9l_{\max} - 9l_{\min}) \geq \frac{1}{72}(4l_{\max} + l_{\min}) \geq \frac{1}{72}L.$$

Case 3: Finally, if $l_{\min} \leq \frac{1}{4}l_{\max}$, then we have

$$\Delta O \geq \frac{1}{12}(l_{\max} - l_{\min}) \geq \frac{1}{72}(6l_{\max} - 6l_{\min}) \geq \frac{1}{72}(4l_{\max} + 2l_{\min}) \geq \frac{1}{72}L.$$

□

5.5 The Proof of Theorem 3.2 for 4-cycles

Lemma 5.10. *If C is 4-cycle, then $2M + 7O \leq 11L$.*

Proof. We again consider several cases.

Case 1: If $l_{\min} > \frac{1}{2}l_{\max}$ then by Lemma 5.3 we have $\Delta O \geq \frac{1}{4}l_{\min}$ and so

$$\Delta O \geq \frac{1}{28}(7l_{\min}) \geq \frac{1}{28}(l_{\min} + 3l_{\max}) \geq \frac{1}{28}L,$$

and the claim follows by Corollary 5.8.

Case 2: If $l_{\min} \leq \frac{1}{2}l_1$, but all down-edges of C are flat then we have by Lemma 5.2 that $\Delta O \geq \frac{1}{4}(l_{\max} - l_{\min})$ and so

$$\Delta O \geq \frac{1}{28}(7l_{\max} - 7l_{\min}) \geq \frac{1}{28}(3l_{\max} + l_{\min}) \geq \frac{1}{28}L.$$

Therefore we only need to consider cases where at least one down-edge of C is steep. This is implicitly assumed in all remaining cases.

Case 3: If $l_1 \geq l_2 \leq l_3 \geq l_4$, i.e. the edges of C are alternating down-up-down-up, then

$$\Delta O = (\Delta o_{41} + \Delta o_{12}) + (\Delta o_{23} + \Delta o_{34}) \geq \frac{1}{6}(l_1 - l_2) + \frac{1}{6}(l_3 - l_4),$$

by Corollary 4.5. We also have

$$M \leq \min(o_{41}, o_{23}) \leq \min(l_1, l_3) \leq \frac{1}{2}(l_1 + l_3).$$

Therefore

$$2M - 7\Delta O \leq l_1 + l_3 + \frac{7}{6}(-l_1 + l_2 - l_3 + l_4) = \frac{1}{6}(-l_1 + 7l_2 - l_3 + 7l_4) \leq \frac{1}{2}(l_1 + l_2 + l_3 + l_4)$$

since $l_2 \leq l_1$ and $l_4 \leq l_3$.

Case 4: If $l_1 \geq l_2 \leq l_3 \leq l_4$, then we have $\Delta O \geq \frac{1}{6}(l_1 - l_2)$ and $M \leq o_{23} \leq l_3$. Therefore

$$2M - 7\Delta O \leq 2l_3 - \frac{7}{6}l_1 + \frac{7}{6}l_2 \leq \left(\frac{1}{2}l_3 + \frac{1}{2}l_4 + l_1\right) - \frac{7}{6}l_1 + \left(\frac{3}{6}l_2 + \frac{4}{6}l_1\right) = \frac{1}{2}L,$$

and the claim follows from Lemma 5.7.

Case 5: If $l_1 \geq l_2 \geq l_3 \leq l_4$, we consider two subcases. Since we excluded Cases 1 and 2, at least one down-edge of C is steep.

Case 5a: If $l_2 \leq \frac{1}{2}l_1$, then $\Delta O \geq \frac{1}{6}(l_1 - l_2)$ and $M \leq o_{34} \leq l_3 + \frac{1}{2}l_4$. Therefore

$$2M - 7\Delta O \leq 2l_3 + l_4 + \frac{7}{6}l_2 - \frac{7}{6}l_1 \leq \left(\frac{1}{2}l_3 + \frac{3}{4}l_1\right) + \left(\frac{1}{2}l_4 + \frac{1}{2}l_1\right) + \left(\frac{3}{6}l_2 + \frac{2}{6}l_1\right) - \frac{7}{6}l_1 \leq \frac{1}{2}L$$

Case 5b: If $l_2 > \frac{1}{2}l_1$ and $l_3 \leq \frac{1}{2}l_2$, then $\Delta O \geq \frac{1}{8}(l_1 - l_3)$ by Lemma 5.2 and $M \leq o_{34} \leq l_3 + \frac{1}{2}l_4$. Therefore

$$2M - 7\Delta O \leq 2l_3 + l_4 - \frac{7}{8}l_1 + \frac{7}{8}l_3 \leq \frac{23}{8}l_3 + l_4 - \frac{7}{8}l_1 \leq \left(\frac{4}{8}l_3 + \frac{4}{8}l_2 + \frac{11}{16}l_1\right) + \left(\frac{1}{2}l_4 + \frac{1}{2}l_1\right) - \frac{7}{8}l_1 \leq \frac{1}{2}L.$$

Case 6: We are left with the case where $l_1 \geq l_2 \geq l_3 \geq l_4$, i.e. C has three down-edges, and at least one of them is steep. We consider three subcases:

Case 6a: If $l_2 \leq \frac{1}{2}l_1$ then similarly to Case 3 we have

$$\Delta O \geq (\Delta o_{41} + \Delta o_{12}) + (\Delta o_{23} + \Delta o_{34}) \geq \frac{1}{6}(l_1 - l_2) + \frac{1}{6}(l_3 - l_4).$$

We also have $M \leq o_{34} \leq l_3 + \frac{1}{2}l_4$. Therefore

$$\begin{aligned} 2M - 7\Delta O &\leq 2l_3 + l_4 + \frac{7}{6}(-l_1 + l_2 - l_3 + l_4) = -\frac{7}{6}l_1 + \frac{7}{6}l_2 + \frac{5}{6}l_3 + \frac{13}{6}l_4 \leq \\ &\leq -\frac{7}{6}l_1 + \left(\frac{3}{6}l_2 + \frac{2}{6}l_1\right) + \left(\frac{3}{6}l_3 + \frac{1}{6}l_1\right) + \left(\frac{3}{6}l_4 + \frac{5}{6}l_1\right) \leq \frac{1}{2}L. \end{aligned}$$

Case 6b: If $l_3 \leq \frac{1}{2}l_2$ then $\Delta O \geq \frac{1}{6}(l_2 - l_3)$ and $M \leq l_3 + \frac{1}{2}l_4$. Therefore

$$\begin{aligned} 2M - 7\Delta O &\leq 2l_3 + l_4 - \frac{7}{6}l_2 + \frac{7}{6}l_3 = -\frac{7}{6}l_2 + \frac{19}{6}l_3 + l_4 \leq -\frac{7}{6}l_2 + \frac{22}{6}l_3 + \frac{1}{2}l_4 \leq \\ &\leq -\frac{7}{6}l_2 + \left(\frac{3}{6}l_1 + \frac{13}{12}l_2 + \frac{3}{6}l_3\right) + \frac{1}{2}l_4 \leq \frac{1}{2}L. \end{aligned}$$

Case 6c: If $l_4 \leq \frac{1}{2}l_3$ then similarly to Case 6a we have

$$\Delta O \geq \frac{1}{6}(l_1 - l_2) + \frac{1}{6}(l_3 - l_4),$$

but this time we use the bound $M \leq o_{41} \leq l_4 + \frac{1}{2}l_1$. We get

$$\begin{aligned} 2M - 7\Delta O &\leq 2l_4 + l_1 + \frac{7}{6}(-l_1 + l_2 - l_3 + l_4) = -\frac{1}{6}l_1 + \frac{7}{6}l_2 - \frac{7}{6}l_3 + \frac{19}{6}l_4 \leq \\ &\leq -\frac{1}{6}l_1 + \left(\frac{3}{6}l_2 + \frac{4}{6}l_1\right) - \frac{7}{6}l_3 + \left(\frac{3}{6}l_4 + \frac{8}{6}l_3\right) \leq \frac{1}{2}L. \end{aligned}$$

□

5.6 The Proof of Theorem 3.2 for 3-cycles

Lemma 5.11. *If C is a 3-cycle then $2M + 7O \leq 11L$.*

Proof. There are essentially two kinds of 3-cycles - ones with (cyclically) increasing l_i , and ones with decreasing l_i .

Case 1: If $l_1 \geq l_2 \geq l_3$ (i.e. l_i are cyclically increasing), then we consider three subcases.

Case 1a: If $l_{\max} < 2l_{\min}$, i.e. $l_1 < 2l_2$, then

$$\Delta O \geq \Delta o_{23} + \Delta o_{31} \geq \left(l_2 - \frac{1}{2}l_3\right) + \left(l_3 - \frac{1}{2}l_1\right) = l_2 + \frac{1}{2}l_3 - \frac{1}{2}l_1 > \frac{1}{2}l_3$$

by Lemma 5.1. We also have $M \leq o_{23} \leq l_3$. Then

$$2M - 7\Delta O \leq 2l_3 - \frac{7}{2}l_3 < 0 \leq \frac{1}{2}L.$$

Case 1b: If $l_1 \geq 2l_3$ then we have

$$\Delta O \geq \Delta o_{12} \geq \frac{1}{6}(l_1 - l_2)$$

and $M \leq o_{23} \leq l_2 + \frac{1}{2}l_3$. Therefore

$$2M - 7\Delta O \leq 2l_2 + l_3 + \frac{7}{6}(-l_1 + l_2) \leq -\frac{7}{6}l_1 + \frac{19}{6}l_2 + l_3 \leq -\frac{7}{6}l_1 + \left(\frac{3}{6}l_2 + \frac{8}{6}l_1\right) + \left(\frac{1}{2}l_3 + \frac{1}{4}l_1\right) \leq \frac{1}{2}L.$$

Case 1c: If $2l_2 \leq l_1 < 2l_3$ then

$$\Delta O \geq \max(\Delta o_{12}, \Delta o_{31}) \geq \max\left(\frac{1}{6}(l_1 - l_2), l_3 - \frac{1}{2}l_1\right)$$

and $M \leq o_{23} \leq l_2 + \frac{1}{2}l_3$. Hence

$$2M - 7\Delta O \leq 2l_2 + l_3 - \left(l_3 - \frac{1}{2}l_1\right) - 6 \cdot \frac{1}{6}(l_1 - l_2) = -\frac{1}{2}l_1 + 3l_2 \leq -\frac{1}{2}l_1 + \left(l_1 + \frac{1}{2}l_2 + \frac{1}{2}l_3\right) \leq \frac{1}{2}L.$$

Case 2: If $l_1 \geq l_2 \geq l_3$ then we consider several subcases. The logic in their ordering is that we are trying to eliminate the easy ones first until only the hardest case remains — one that is actually tight.

Case 2a: If $l_1 < 2l_3$, i.e. $l_{\max} < 2l_{\min}$, then by Lemma 5.3 we have $\Delta O \geq \frac{1}{4}l_3$. We also have $M \leq \min(o_{23}, o_{31}) \leq \min(l_2 + \frac{1}{2}l_3, l_1)$. Therefore

$$2M - 7\Delta O \leq \frac{1}{2}\left(l_2 + \frac{1}{2}l_3\right) + \frac{3}{2}l_1 - \frac{7}{4}l_3 = \frac{3}{2}l_1 + \frac{1}{2}l_2 - \frac{3}{2}l_3 \leq \frac{1}{2}L.$$

Case 2b: If $l_1 \geq 2l_3$, but both down-edges are flat, then by Lemma 5.2 we have $\Delta O \geq \frac{1}{4}(l_1 - l_3)$. Using the same bound on M as in the previous case, we obtain

$$2M - 7\Delta O \leq \frac{1}{2}\left(l_2 + \frac{1}{2}l_3\right) + \frac{3}{2}l_1 - \frac{7}{4}(l_1 - l_3) = -\frac{1}{4}l_1 + \frac{1}{2}l_2 + 2l_3 \leq \frac{1}{2}L.$$

We are left with the case where at least one down-edge of C is steep.

Case 2c: If $l_2 \leq \frac{1}{2}l_1$ then we have $\Delta O \geq \frac{1}{6}(l_1 - l_2)$ by Corollary 4.5. We also have $M \leq o_{23} \leq l_2 + \frac{1}{2}l_3$. Hence

$$2M - 7\Delta O \leq 2l_2 + l_3 - \frac{7}{6}l_1 + \frac{7}{6}l_2 = -\frac{7}{6}l_1 + \frac{19}{6}l_2 + l_3 \leq -\frac{7}{6}l_1 + \left(\frac{3}{6}l_2 + \frac{8}{6}l_1\right) + \left(\frac{1}{2}l_3 + \frac{1}{4}l_1\right) \leq \frac{1}{2}L.$$

Case 2d: If $l_2 > \frac{1}{2}l_1$ and $2l_3 \leq l_2 < \frac{5}{2}l_3$ then we have

$$\Delta O \geq \Delta o_{12} + \Delta o_{23} \geq \Delta \alpha_2 + \Delta o_{23} \geq \left(\frac{3}{2}l_2 + \frac{1}{2}l_3\right) - (2l_2 - l_3) = \frac{3}{2}l_3 - \frac{1}{2}l_2$$

by Lemma 4.4, and we also have

$$\Delta O \geq \Delta o_{12} \geq \frac{1}{2}(l_1 - l_2).$$

Joining these two bounds with $M \leq o_{31} \leq l_3 + \frac{1}{2}l_1$ gives

$$2M - 7\Delta O \leq 2l_3 + l_1 - 6\left(\frac{3}{2}l_3 - \frac{1}{2}l_2\right) - \frac{1}{2}(l_1 - l_2) = \frac{1}{2}l_1 + \frac{7}{2}l_2 - 7l_3 \leq \frac{1}{2}l_1 + \left(\frac{1}{2}l_2 + \frac{15}{2}l_3\right) - 7l_3 = \frac{1}{2}L.$$

Case 2e: Finally if $l_2 > \frac{1}{2}l_1$ and $l_2 \geq \frac{5}{2}l_3$ then we have

$$\Delta O \geq \Delta o_{12} + \Delta o_{23} \geq \Delta \alpha_2 + \Delta o_{23} \geq \left(\frac{3}{2}l_2 + \frac{1}{2}l_3\right) - (l_2 + l_3 + \alpha_3) \geq \frac{1}{2}l_2 - l_3$$

by Lemma 4.4. We now proceed similarly to the previous case:

$$2M - 7\Delta O \leq 2l_3 + l_1 - 6\left(\frac{1}{2}l_2 - l_3\right) - \frac{1}{2}(l_1 - l_2) \leq \frac{1}{2}l_1 - \frac{5}{2}l_2 + 8l_3 = \frac{1}{2}L - 3l_2 + \frac{15}{2}l_3 \leq \frac{1}{2}L.$$

□

5.7 The Proof of Theorem 3.2 for 2-cycles

Before we proceed with the case of 2-cycles, we need an additional technical lemma.

Lemma 5.12. *If $l_1 \geq 2l_2$ then $\Delta o_{12} + \Delta o_{21} \geq \frac{1}{2}l_2$.*

Proof. If $o_{12} \leq l_1$ then clearly $\Delta O \geq \Delta o_{12} \geq \frac{1}{2}l_2$. Hence we can assume $o_{12} \geq l_1$. Note that this means that l_1 is not a multiple of l_2 , since w_1 is primitive.

Assume w.l.o.g. that w_2 is its maximal rotation and let $k \geq 2$ be such, that $kl_2 < l_1 < (k+1)l_2$. Since $o_{12} \geq l_1$, by Lemma 4.2 we get $i_{\min}(w_{12}) = \alpha_2 + 1$ and $i_{\max}(w_{12}) \geq kl_2 + 1$. This means that $|p_{\min}(w_1)| \geq kl_2 - \alpha_2 = (k-1)l_2 + (l_2 - \alpha_2) \geq \frac{3}{2}l_2$ and $|p_{\max}(w_1)| \leq l_1 - kl_2 + \alpha_2 < \frac{3}{2}l_2$. Therefore, w_1 is its maximal rotation as well.

Since $i_{\max}(w_{12}) \geq kl_2 + 1$ and l_2 does not divide l_1 , we know $p_{\max}(w_1) = wp_{\max}(w_2)$, where $w = w_{12}[i_{\max}(w_{12}), l_1]$. Note that $|w| < |w_2|$ and w is a prefix of w_2 (because it is an initial segment of a maximal rotation of w_1).

We will show that $o_{21} < |p_{\max}(w_1)| = \alpha_1$, which implies the claim of the lemma. Assume the opposite, i.e. $o_{21} \geq \alpha_1$. Then $o_{21}[1, \alpha_1] = p_{\max}(w_1) = wp_{\max}(w_2)$. By Lemma 2.2 this can only happen if w_2 is aligned with position $|w| + 1$ of ov_{21} . But then w is a suffix of w_2 , and since it is also a prefix of w_2 we get a contradiction with Lemma 2.3. □

Lemma 5.13. *If C is a 2-cycle, then $2M + 7O \leq 11L$.*

Proof. We consider three cases.

Case 1: If $l_1 < 2l_2$, i.e. the down-edge of C is flat, then we have

$$\Delta O \geq \max\left(\frac{1}{2}(l_1 - l_2), l_2 - \frac{1}{2}l_1\right)$$

by Corollary 4.5 and Lemma 5.1. We also have $M \leq o_{21} \leq l_1$, and so

$$2M - 7\Delta O \leq 2l_1 - 5 \cdot \frac{1}{2}(l_1 - l_2) - 2\left(l_2 - \frac{1}{2}l_1\right) = \frac{1}{2}L.$$

Case 2: If $2l_2 \leq l_1 < 3l_2$ then we have $\Delta O \geq \frac{1}{2}l_2$ by Lemma 5.12. This easily gives our main claim since using $M \leq o_{21} \leq l_2 + \frac{1}{2}l_1$ we have

$$2M - 7\Delta O \leq 2l_2 + l_1 - \frac{7}{2}l_2 = l_1 - \frac{3}{2}l_2 \leq \left(\frac{1}{2}l_1 + \frac{3}{2}l_2\right) - \frac{3}{2}l_2 \leq \frac{1}{2}L.$$

Case 3: If $3l_2 \leq l_1$ then by Corollary 4.5 we have $\Delta O \geq \frac{1}{4}(l_1 - l_2)$ and together with $M \leq o_{21} \leq l_2 + \frac{1}{2}l_1$ we get

$$2M - 7\Delta O \leq 2l_2 + l_1 - \frac{7}{4}l_1 + \frac{7}{4}l_2 = -\frac{3}{4}l_1 + \frac{15}{4}l_2 = -\frac{3}{4}l_1 + \frac{5}{4}l_1 \leq \frac{1}{2}L.$$

□

6 Tight examples

6.1 Tightness of Theorem 3.2

We will now show that Theorem 3.2 is essentially tight. To this end, we give two examples of cycles in the overlap graph, for which $2M + 7O = 11L - O(1)$. Note that by increasing the lengths of the strings in these cycles we can get $\frac{2M+7O}{11L} \rightarrow 1$.

Example 6.1. Let $w_1 = ba^kb|a^{k+1}ba^{k+1}$ and $w_2 = a^{k+1}|ba^kb$ (we use the symbol $|$ to mark the border between p_{\max} and p_{\min}). Here $l_1 = 3k + 5$, $l_2 = 2k + 3$, so $L = 5k + 8$.

Now, let $x_i = (w_i^2)[1, 2l_i - 1]$ for $i = 1, 2$. Note that all w_i are nice words and every x_i is a w_i -word.

We have $o_{12} = 4k + 5$ and $o_{21} = 3k + 4$, so $O = 7k + 9$ and $M = 3k + 4$. Note that $2M + 7O = 6k + 8 + 49k + 63 = 55k + 71 = 11L - O(1)$.

Example 6.2. Let $w_1 = ba^nba^{n+1}ba^n|a^{n+1}ba^{n+1}ba^{n+1}$, $w_2 = a^{n+1}ba^{n+1}|ba^nba^{n+1}ba^n$, $w_3 = a^{n+1}|ba^n$. We have $l_1 = 6n + 10$, $l_2 = 5n + 8$, $l_3 = 2n + 3$, so $L = 13n + 21$.

Now, let $x_1 = (w_1^2)[1, 2l_1 - 1]$, $x_2 = (w_2^3)[1, 2l_2 + \alpha_2 - 1]$ and $x_3 = (w_3^4)[1, 4l_3 - 1]$. Note that all w_i are nice words and every x_i is a w_i -word.

We have $o_{12} = 8n + 12$, $o_{23} = 6n + 8$ and $o_{31} = 5n + 7$, so $O = 19n + 27$ and $M = 5n + 7$. Note that $2M + 7O = 10n + 14 + 133n + 189 = 11L - O(1)$.

We will now show that the bound we give on $\min(M, (1 - c)O)$ in Corollary 3.4 is also essentially tight.

Consider a cycle cover \mathcal{C} in the overlap graph, composed of two collections of cycles: \mathcal{C}_2 consisting of 2-cycles of the form described in Example 6.1, and \mathcal{C}_3 consisting of 3-cycles described in Example 6.2 (note that these cycles need to use different n so that their vertices are non-equivalent).

Let L_2, L_3 be the total length of the periods of the strings in the cycles of \mathcal{C}_2 and \mathcal{C}_3 , respectively. Let O_2 be the sum of all overlaps on the cycles in \mathcal{C}_2 and let M_2 be the sum of smallest overlaps for each cycle in \mathcal{C}_2 . Similarly define O_3 and M_3 for \mathcal{C}_3 . Finally let $L = L_2 + L_3$, $O = O_2 + O_3$ and $M = M_2 + M_3$.

Note that to make the analysis in Corollary 3.4 tight we only need to make $M = (1 - c)O$, since we already have $2M + 7O = 11L - O(1)$. Since $M_2 \sim \frac{3}{7}O_2$ and $M_3 \sim \frac{5}{19}O_3$, this can be done by adjusting the balance between L_2 and L_3 , provided that $c \in [\frac{4}{7}, \frac{14}{19}]$. The current best approximation ratio of $\frac{2}{3}$ for MAX-ATSP-PATH sits well within this interval.

6.2 The Greedy Algorithm

Recall the greedy algorithm, which picks two strings with the largest overlap and combines them together until a single string remains. The bounds in Breslauer et al. can be used to improve the analysis of this algorithm, as shown by Kaplan et al. [11]. It is natural to ask whether our bounds can be used in a similar fashion. Unfortunately, it seems that there is no simple way to do this. In their analysis, Kaplan et al. require a good bound on the total overlap of a (possibly) long path of strings in the overlap graph. As it turns out, in this case the overlap can actually approach the bound of $\frac{3}{2} \sum_i l_i$ arbitrarily close, as can be seen in the following example.

Example 6.3. For any $k \geq 1$ let $w_{2k} = b^k | a^k$ and $w_{2k-1} = a^{k-1} | b^k$. Also, let $x_{2k} = b^k a^k b^k a^{k-1}$ and $x_{2k-1} = a^{k-1} b^k a^{k-1} b^{k-1}$. Note that all w_i are nice words and every x_i is a w_i -word.

Consider $S = \{x_3, x_4, \dots, x_n\} = \{ab^2ab, b^2a^2b^2a, a^2b^3a^2b^2, b^3a^3b^3a^2, \dots\}$ and the path $x_n \rightarrow x_{n-1} \rightarrow \dots \rightarrow x_3$ in the overlap graph of S . It is easy to verify that $o_{i+1,i} = \lfloor \frac{3i}{2} \rfloor$. Therefore, the total overlap of the path is approximately $\frac{3}{4}n^2$, and $\sum_{i=3}^n l_i \sim \frac{1}{2}n^2$.

This, of course, does not rule out using our results to improve the analysis of the greedy algorithm. However, any such result requires some additional insight.

7 Acknowledgements

The author would like to thank Anupam Gupta and Piotr Sankowski for valuable suggestions that helped to improve the presentation of the paper, and Jakub Radoszewski for his advice in matters of stringology.

References

- [1] Chris Armen and Clifford Stein. Improved length bounds for the shortest superstring problem. In *WADS*, pages 494–505, 1995.
- [2] Chris Armen and Clifford Stein. A $2 \frac{2}{3}$ -approximation algorithm for the shortest superstring problem. In *CPM*, pages 87–101, 1996.

- [3] Arash Asadpour, Michel X. Goemans, Aleksander Madry, Shayan Oveis Gharan, and Amin Saberi. An $O(\log n / \log \log n)$ -approximation algorithm for the asymmetric traveling salesman problem. In *SODA*, pages 379–389, 2010.
- [4] Avrim Blum, Tao Jiang, Ming Li, John Tromp, and Mihalis Yannakakis. Linear approximation of shortest superstrings. *J. ACM*, 41(4):630–647, 1994.
- [5] Dany Breslauer, Tao Jiang, and Zhigen Jiang. Rotations of periodic strings and short superstrings. *J. Algorithms*, 24(2):340–353, 1997.
- [6] Maxime Crochemore and Dominique Perrin. Two-way string matching. *J. ACM*, 38(3):651–675, 1991.
- [7] Artur Czumaj, Leszek Gasieniec, Marek Piotrów, and Wojciech Rytter. Sequential and parallel approximation of shortest superstrings. *J. Algorithms*, 23(1):74–100, 1997.
- [8] John Gallant, David Maier, and James A. Storer. On finding minimal length superstrings. *J. Comput. Syst. Sci.*, 20(1):50–58, 1980.
- [9] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [10] Haim Kaplan, Moshe Lewenstein, Nira Shafrir, and Maxim Sviridenko. Approximation algorithms for asymmetric tsp by decomposing directed regular multigraphs. *J. ACM*, 52(4):602–626, 2005.
- [11] Haim Kaplan and Nira Shafrir. The greedy algorithm for shortest superstrings. *Inf. Process. Lett.*, 93(1):13–17, 2005.
- [12] S. Rao Kosaraju, James K. Park, and Clifford Stein. Long tours and short superstrings. In *FOCS*, pages 166–177, 1994.
- [13] A.M. Lesk. *Computational molecular biology: sources and methods for sequence analysis*. Oxford University Press, 1988.
- [14] M. Li. Towards a dna sequencing theory. In *FOCS*, pages 125–134, 1990.
- [15] M. Lothaire. *Combinatorics on Words*. Cambridge Mathematical Library. Cambridge University Press, 1997.
- [16] Katarzyna E. Paluch, Khaled M. Elbassioni, and Anke van Zuylen. Simpler approximation of the maximum asymmetric traveling salesman problem. In *STACS*, pages 501–506, 2012.
- [17] Hannu Peltola, Hans Söderlund, Jorma Tarhio, and Esko Ukkonen. Algorithms for some string matching problems arising in molecular genetics. In *IFIP Congress*, pages 59–64, 1983.
- [18] James A. Storer. *Data Compression: Methods and Theory*. Computer Science Press, 1988.
- [19] Z. Sweedyk. A $2\frac{1}{2}$ -approximation algorithm for shortest superstring. *SIAM J. Comput.*, 29(3):954–986, 1999.

- [20] Jorma Tarhio and Esko Ukkonen. A greedy approximation algorithm for constructing shortest common superstrings. *Theor. Comput. Sci.*, 57:131–145, 1988.
- [21] Shang-Hua Teng and F. Frances Yao. Approximating shortest superstrings. *SIAM J. Comput.*, 26(2):410–417, 1997.
- [22] Jonathan S. Turner. Approximation algorithms for the shortest common superstring problem. *Inf. Comput.*, 83(1):1–20, 1989.
- [23] Virginia Vassilevska. Explicit inapproximability bounds for the shortest superstring problem. In *MFCS*, pages 793–800, 2005.
- [24] Michael S. Waterman. *Introduction to computational biology - maps, sequences, and genomes: interdisciplinary statistics*. CRC Press, 1995.
- [25] Y.Césari and M. Vincent. Une caractérisation des mots périodiques. *C. R. Acad. Sci. Paris Sér. A*, 286:1175–1177, 1978.